METHOD AND SYSTEM FOR RETRIEVING INFORMATION, AND COMPUTER PRODUCT

FIELD OF THE INVENTION

The present invention in general relates to an information retrieving system that retrieves design/manufacturing information of a great variety of file formats registered in a plurality of web servers, by using an index server disposed at the center. More particularly, this invention relates to a method and system for retrieving information capable of efficiently automatically registering information registered in a plurality of web servers into an index server.

BACKGROUND OF THE INVENTION

Conventionally, an enterprise has had a function that integrally manages a company design information system. When each business department independently registers design/manufacturing information relating to the hardware and firmware of a product into a server, this design/manufacturing information can be provided to users beyond the fence of the business department.

When an information-retrieving user wants to retrieve necessary design information, the user searches this information based on one of the following methods. The user

(1) utilizes an URL list table that shows servers and business departments in which the necessary design/manufacturing information are stored. The user (2) searches the information by following the link. The user (3) searches the information in a homepage of a likely business department. The user (4) uses a retrieval tool provided by the homepage of each business department.

However, there are a great variety of servers of design functions and manufacturing functions that are to be searched. Further, there are alterations, deletions and additions of URLs in respective servers for the sake of convenience of the function of managing each server. Therefore, it has not always been possible for the information-retrieving users to catch this information.

Furthermore, there are a variety of data formats and file formats of information that are registered in each server. Therefore, the information-retrieving users have not been able to understand who has registered what information and when such information has been registered, at all. Moreover, in many cases, the information-retrieving users suddenly want to obtain design/manufacturing information. Therefore, it is not efficient to ask the information-retrieving users to understand the location of this information.

SUMMARY OF THE INVENTION

It is an object of this invention to provide an information retrieving system capable of efficiently automatically registering information registered in a plurality of web servers into an index server, thereby to enable information-retrieving users to promptly and efficiently retrieve the information. It is another object of this invention to provide a computer readable recording medium that stores a computer program which when executed realizes the method according to the present invention.

The information retrieving system according to one aspect of this invention comprises a transmitting unit that transmits an abstract and storage location information of design/manufacturing information registered in the plurality of web servers, to an index server during an idle time of the web servers; and an automatic registering unit that automatically registers an abstract and storage location information transmitted from web servers by the transmitting unit, into the index server.

According to the above-mentioned aspect, an abstract and storage location information of design/manufacturing information registered in the plurality of web servers, are transmitted to an index server during an idle time of the web servers. Further, the abstract and storage location information that have been transmitted are automatically

registered into the index server. Therefore, it is possible to efficiently automatically register the information registered in the plurality of web servers into the index server, thereby to enable information-retrieving users to promptly and efficiently retrieve the information.

The information retrieving method according to another aspect of this invention comprises a transmission step of transmitting an abstract and storage location information of design/manufacturing information registered in the plurality of web servers, to an index server during an idle time of the web servers; and an automatic registration step of automatically registering an abstract and storage location information transmitted from web servers by the transmitting unit, into the index server.

According to the above-mentioned aspect, an abstract and storage location information of design/manufacturing information registered in the plurality of web servers, are transmitted to an index server during an idle time of the web servers. Further, the abstract and storage location information that have been transmitted are automatically registered into the index server. Therefore, it is possible to efficiently automatically register the information registered in the plurality of web servers into the index server, thereby to enable information-retrieving users to promptly and efficiently retrieve the information.

Other objects and features of this invention will become apparent from the following description with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram showing a system structure of an information retrieving system relating to an embodiment of the present invention;

Fig. 2 is a diagram for explaining a generation of an abstract from a structure diagram of an ICAD-MX format;

Fig. 3 is a diagram for explaining a registration starting operation by a user and a registration processing by a register processing agent;

Fig. 4 is a diagram showing one example of an index menu of the information retrieving system shown in Fig. 1;

Fig. 5 is a flowchart showing a procedure of a register processing agent shown in Fig. 1;

Fig. 6 is a diagram showing one example of a data structure stored in a repository;

Fig. 7 is a flowchart showing a procedure of a register processing agent when a registration document consists of a drawing file of the ICAD-MX or a document file;

Fig. 8 is a flowchart showing a procedure of a registration accept processing agent shown in Fig. 1;

Fig. 9 is a flowchart showing a procedure when the

information retrieving system shown in Fig. 1 retrieves an abstract;

Fig. 10 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves a keyword;

Fig. 11 is a flowchart showing a procedure of an OVI search shown at step S904 in Fig. 9 and at step S1012 in Fig. 10 respectively;

Fig. 12 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves an operation process;

Fig. 13 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves a rule;

Fig. 14A and Fig. 14B are diagrams showing one example of a display screen when the information retrieving system shown in Fig. 1 retrieves a keyword;

Fig. 15A and Fig. 15B are diagrams showing one example of a display screen when the information retrieving system shown in Fig. 1 retrieves a keyword;

Fig. 16A and Fig. 16B are diagrams showing one example of a display screen when the information retrieving system shown in Fig. 1 retrieves a life cycle;

Fig. 17 is a diagram showing one example of a display screen when the information retrieving system shown in Fig.

1 retrieves a life cycle;

Fig. 18A to Fig. 18C are diagrams showing one example of a display screen when the information retrieving system shown in Fig. 1 retrieves an operation process association;

Fig. 19A and Fig. 19B are diagrams showing one example of a display screen when the information retrieving system shown in Fig. 1 retrieves an overview index; and

Fig. 20 is a diagram showing one example of a hierarchical structure of documents and drawings.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Preferred embodiments of an information retrieving system relating to the present invention will be explained in detail below with reference to the accompanying drawings. For the sake of convenience of the explanation, a case of using two web servers will be explained below.

First, a system structure of an information retrieving system and architecture of the whole system relating to the present embodiment will be explained. Fig. 1 is a block diagram showing a system structure of an information retrieving system relating to the present embodiment. When a user wants to retrieve design/manufacturing information of a great variety of file formats registered in the plurality of web servers 110 or 120, the information retrieving system shown in Fig. 1 works as follows. The information retrieving

system enables the user to promptly retrieve location information of each design/manufacturing information in a web server 110 or 120 by referring to an index server 130 disposed at the center of the system.

In order to make it possible to achieve this information retrieval, the information retrieving system employs the following architecture. First, (1) a simple icon is provided in advance. When an information-retrieving user drops design/manufacturing information that the user wants to register onto this icon, an ontological document hierarchical structure is provided.

Further, (2) a method of generating a multimedia abstract is provided, so that a general-purpose abstract that is beyond the kinds of document/drawing information preparation tools is automatically generated. Further, (3) an agent for registering the design/manufacturing information is used to improve web traffics.

Further, (4) a repository for the abstract of the design/manufacturing information and the storage location information capable of bridging between various servers is provided. The agent carries out the information communications. Further, (5), retrieving means according to the operation process, for example, is provided, thereby to improve the retrieving method.

A system structure of the information retrieving

system relating to the present embodiment will be explained next. As shown in Fig. 1, this information retrieving system has the plurality of web servers 110 and 120, an index server 130, and a component name server 140, connected via a router 150.

The web servers 110 and 120 are the servers that provide the stored design information to users via a WEB SERVER (World Wide Web) as a distributed system on the Internet. Each of the web servers 110 and 120 is built in with a register processing agent 111.

The register processing agent 111 has an abstract generation (ABSGEN) 112 that is started during an idle time after a document has been registered, and that automatically generates a registration document. The register processing agent 111 further has a location generator (LOCGEN) 113 that generates an URL as a storage location of a registered document, and a transmission (TRANSM) 114 that transfers a storage location and an abstract of a registered document to the index server 130 during an idle time.

The register processing agent 111 stores the abstract and the storage information of the registered information into a repository 115, and transfers the abstract and the storage information to the index server 130 by utilizing an idle time of the web server 110 or 120.

The index server 130 is built in with a registration

accept processing agent 131 for carrying out a registration acceptance processing. This registration accept processing agent 131 has an overview index register (OVIREG) 132 that stores information relating to a registered document into a repository 135 during an idle time. The registration accept processing agent 131 further has an ontological restructuring (ONTGEN) 133 that restructures the consistency of the ontology of the hierarchical structure, and an accept signal generator (ACPGEN) 134 that posts to the web server a fact that information has been registered into the index server 130.

The component name server 140 is a server that stores product development codes, a list of terms, a list of abbreviations, units, software manuals, hardware manuals, personnel and organization information, E-mail information and the like. When an information retrieval request has been received from an information-retrieving user, the component name server 140 provides various kinds of information to the index server 130.

Next, the generation of an abstract from multimedia by the ABSGEN 112 shown in Fig. 1 will be explained. The information provided by the web servers 110 and 120 includes a great variety of design/manufacturing information such as characters, drawings and images.

In order to retrieve these information, it is essential

to utilize an abstract that has been expressed by a standard language (XML). Therefore, as shown in Fig. 2, the ABSGEN 112 collects meta information of the ICAD-MX from a structure diagram FJ999 of the ICAD-MX format, thereby to generate an abstract.

Specifically, the ABSGEN 112 automatically recognizes that the file format has been designed by the ICAD-MX tool, converts the ICAD-MX file into a text file as a standard document, and searches and collects a character string given to the structure portion. This drawing number is a structured drawing number system that has ontological characteristics of upper-layer and lower-layer structures. Therefore, the ABSGEN 112 enriches the contents of the own abstract by citing the diagram of the high and low hierarchy.

Next, a registration procedure by a user and a registration processing by the register processing agent 111 will be explained. Fig. 3 is a diagram for explaining a registration starting operation by the user and the registration processing by the register processing agent.

As shown in Fig. 3, first, a file list is displayed on the display screen, and a registration processing folder for the processing to be carried out by the register processing agent 111 is displayed on the display screen. Then, in the case of registering an FJ999 server structure diagram, for example, this structure diagram is dragged to

the registration processing folder and drops it.

When the registration has been carried out in the manner as described above, the register processing agent 111 is put into a queue, so that the register processing agent 111 is started during an idle time. Specifically, first, the ABSGEN 112 automatically generates an abstract, and then the LOCGEN 113 generates storage information. Thereafter, the TRANSM 114 that transfers the abstract and the storage information is put into a queue.

When an idle time has been found, the registration accept processing agent 131 is started. Then, the OVIREG 132 stores the abstract and the storage information into the repository 135 of the index server 130, and the ONTGEN 133 restructures the ontology hierarchical structure. Then, the ACPGEN 134 posts to the registration originator web server 110 or 120 a reception signal that shows that the abstract and the storage information have been registered into the index server 130. In this case, communications are carried out between the repository 115 held by the web server 110 or 120 and the repository 135 held by the index server 130 shown in Fig. 1.

Next, the index menu of the information retrieving system shown in Fig. 1 will be explained. Fig. 4 is a diagram that shows one example of an index menu of the information retrieving system shown in Fig. 1.

The index menu shown in Fig. 4 is a menu that displays a layout of drawings for each of the design and manufacturing processes of the FJ999 server. When any one desired item is selected from this menu with a mouse, a relevant document is output at high speed.

Specifically, the design and manufacturing processes are classified into "product development plan", "design specification", and "actual design". Then, the product development plan is corresponded to the FJ997, FJ998 or FJ999, the design specification is corresponded to a power source specification, a server specification, and a network specification, and the actual design is corresponded to a power source diagram and a structure diagram.

A description by the XML becomes as follows.

<xml> <product development plan> FJ999 </product development plan>

<design specification> Server specification </design specification>

<actual design> structure diagram </actual design> <xml>

A procedure of the register processing agent 111 shown in Fig. 1 will be explained next. Fig. 5 is a flowchart showing a procedure of the register processing agent shown in Fig. 1. As shown in Fig. 5, the register processing agent 111 first carries out a text conversion processing for

converting a registered document into a text (step S501), and then converts the text into an XML language, and tags each word (step S502).

Then, the register processing agent 111 files the XML language into a folder of the ontology corresponding to the drawing number (step S503), and stores a result into the repository of the corresponding diagram (step S504). For example, when the repository structure of the drawing number is "CAxx-yyy-zzzz" (where xx denotes a directory, yyy denotes a first serial number, and zzzz denotes a second serial number), the register processing agent 111 files the XML language into a folder of a corresponding hierarchy shown by this xx.

Thereafter, the register processing agent 111 stores a URL generated by the LOCGEN 113 into the same repository (step S505), and stores the abstract generated by the ABSGEN 112 into the same repository (step S506). Based on this arrangement, the repository stores data consisting of a web server ID 601, a URL 602, a file name 603, an abstract 604 of the text, and an abstract 605 of the XML, as shown in Fig. 6.

Next, a procedure of the register processing agent 111 when a registered document consists of a drawing file of the ICAD-MX or a document file will be explained. Fig. 7 is a flowchart showing a procedure of the register

processing agent 111 when a registration document consists of a drawing file of the ICAD-MX or a document file.

As shown in Fig. 7, the register processing agent 111 decides a type of a drawing or document file (step S701). When the file is the ICAD-MX (Yes at step S702), the register processing agent 111 reads characters from the contents of the drawings (step S703). When the file is not the ICAD-MX (No at step S702), the register processing agent 111 collects a text sentence corresponding to each document file (step S704).

Thereafter, the register processing agent 111 generates an abstract from the text sentence (step S705), converts this abstract into an XML format (step S706), and adds the registration ID and the URL of the web server (steps S707 and S708). During an idle time, the register processing agent 111 transfers the abstract to the index server 130 (step S709).

Next, a procedure of the registration accept processing agent 131 shown in Fig. 1 will be explained. Fig. 8 is a flowchart showing a procedure of the registration accept processing agent 131 shown in Fig. 1. The registration accept processing agent 131 first obtains a drawing name or a document name (step S801), and then confirms whether there is a CA diagram or not (step S802). When there is a CA drawing number (Yes at step S802), the registration

accept processing agent 131 stores the CA drawing number into a folder of the repository corresponding to the drawing number (step S810).

On the other hand, when there is no CA drawing number (No at step S802), the registration accept processing agent 131 confirms whether there is a development code name or not (step S803). When there is a development code name (Yes at step S803), the registration accept processing agent 131 obtains a drawing number from this development code name (step S804), and stores the drawing number into a folder of the repository corresponding to the drawing number (step S810).

On the other hand, when there is no development code name (No at step S803), the registration accept processing agent 131 confirms whether there is a device number or not (step S805). When there is a device number (Yes at step S805), the registration accept processing agent 131 obtains a drawing number from this device number (step S806), and stores the drawing number into a folder of the repository corresponding to the drawing number (step S810).

On the other hand, when there is no device number (No at step S805), the registration accept processing agent 131 confirms whether there is a machine type or a series name or not (step S807). When there is a machine type or a series name (Yes at step S807), the registration accept processing

agent 131 obtains a drawing number from the machine type or the series name (step S808). Then, the registration accept processing agent 131 stores the drawing number into a folder of the repository corresponding to the drawing number (step S810).

On the other hand, when there is no machine type or series name (No at step S807), the registration accept processing agent 131 obtains a drawing number from the abstract of the drawing or document (step S808). Then, the registration accept processing agent 131 stores the drawing number into a folder of the repository corresponding to the drawing number (step S810).

Next, a procedure for retrieving an abstract by the information retrieving system shown in Fig. 1 will be explained. Fig. 9 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves an abstract.

As shown in Fig. 9, for retrieving an abstract, the information retrieving system first reads a keyword (step S901), and searches a relevant drawing number element from the keyword (step S902). When there is a relevant drawing number element (Yes at step S903), the information retrieving system carries out an overview index search (an OVI search) to be described later (step S904), then searches a repository (step S905), and displays a list of relevant drawing numbers

(step S906). When there is no relevant drawing number element (No at step S903), the information retrieving system proceeds straight to step S907.

Then, the information retrieving system confirms whether there is a suitable word in an optional abstract or not (step S907). When there is a suitable word (Yes at step S907), the information retrieving system reads the abstract of a relevant document from the repository, and displays the abstract (step S908). When there is no suitable word (No at step S907), the information retrieving system displays an abstract having a relevant tag (step S909).

Next, a procedure for retrieving a keyword by the information retrieving system shown in Fig. 1 will be explained. Fig. 10 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves a keyword.

As shown in Fig. 10, when the information retrieving system has accepted a request for retrieving a keyword (step S1001), the information retrieving system decodes this keyword (step S1002). When the keyword is a machine type name (Yes at step S1003), the information retrieving system obtains a drawing number from this machine type name (step S1004), searches the repository (step S1011), and carries out an overview index search (an OVI search) to be described later (step S1012).

On the other hand, when the keyword is not a machine type name (No at step S1003), and the keyword is a development code name (Yes at step S1005), the information retrieving system obtains a drawing number from this development code name (step S1006). Then, the information retrieving system searches the repository (step S1011), and carries out the OVI search (step S1012).

On the other hand, when the keyword is not a development code name (No at step S1005), and the keyword is a series model name (Yes at step S1007), the information retrieving system obtains a drawing number from this series model name (step S1008). Then, the information retrieving system searches the repository (step S1011), and carries out the OVI search (step S1012).

On the other hand, when the keyword is not a series model name (No at step S1007), and the keyword is a purchased product (Yes at step S1009), the information retrieving system obtains a drawing number from this purchased product (step S1010). Then, the information retrieving system searches the repository (step S1011), and carries out the OVI search (step S1012). When the keyword is not a purchased product either (No at step S1009), the information retrieving system finishes the processing.

Next, the procedure of the OVI search shown at step S904 in Fig. 9 and at step S1012 in Fig. 10 respectively

will be explained. Fig. 11 is a flowchart showing a procedure of the OVI search shown at step S904 in Fig. 9 and at step S1012 in Fig. 10 respectively.

As shown in Fig. 11, in the OVI search, when a CA drawing number is to be searched (Yes at step S1101), the information retrieving system makes a decision about in what hierarchical layer from a master drawing number the repository exists (step S1102). The information retrieving system decides a hierarchical layer from the first serial number and the second serial number (steps S1103 and S1104), and finds a drawing number folder of an object (step S1105). When a CA drawing number is not to be searched (No at step S1101), the information retrieving system finishes the processing.

Next, a procedure for retrieving an operation process by the information retrieving system shown in Fig. 1 will be explained. Fig. 12 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves an operation process.

As shown in Fig. 12, for retrieving an operation process, the information retrieving system records an operation process root (step S1201), and confirms whether there exists a past root or not (step S1202).

When there does not exit a past root (No at step S1202), the information retrieving system adds an operation process retrieval number (step S1205), and finishes the processing.

When there exists a past root (Yes at step S1202), the information retrieving system places a weight on a retrieval number corresponding to the operation process (step S1203), and then shows necessary information menus in the order of high weight at each point of activity (step S1204).

Next, a procedure for retrieving a rule by the information retrieving system shown in Fig. 1 will be explained. Fig. 13 is a flowchart showing a procedure when the information retrieving system shown in Fig. 1 retrieves a rule. In this case, it is assumed that the information retrieving system retrieves a rule of "IF A=x then DO XXX".

As shown in Fig. 13, for retrieving a rule, the information retrieving system reads a first rule statement (step S1301), and obtains a subject A assigned in this rule statement (step S1302).

Thereafter, the information retrieving system obtains an object x assigned in the rule statement (step S1303), and stores the weight of a document or a drawing number into the repository when the subject A = object x (step S1304).

Next, a procedure for retrieving a keyword by the information retrieving system shown in Fig. 1 will be explained. Fig. 14A, Fig. 14B, Fig. 15A and Fig. 15B are diagrams showing one example of a display screen when the information retrieving system shown in Fig. 1 retrieves a keyword. In this case, it is assumed that the information

retrieving system retrieves a design review material of a development plan for a device called "SYDNEY Model2000".

As shown in Fig. 14A, a search starting is instructed by inputting "SYDNEY" as a first keyword, "Model2000" as a second keyword, and "Development plan" as a third keyword of a window 1401 respectively, and by selecting "AND" as a search condition. Then, a result of the searching is obtained.

When a list display is selected on this window 1401, a window 1402 that includes a result of the searching arrayed in the order of high weight is displayed as shown in Fig. 14B. Therefore, "SYDNEY Model2000 Development plan" is selected as a first category, and a keyword of "Design review" is assigned on the window 1402, and a search starting is instructed. Then, a window 1403 shown in Fig. 15A is displayed.

Therefore, "SYDNEY Model2000 Design review" is selected on this window 1403, and a search starting is instructed. As a result, a window 1404 that displays a corresponding design review is displayed on the screen as shown in Fig. 15B.

Next, a procedure for retrieving a life cycle by the information retrieving system shown in Fig. 1 will be explained. Fig. 16A, Fig. 16B and Fig. 17 are diagrams showing one example of a display screen when the information

retrieving system shown in Fig. 1 retrieves a life cycle. In this case, it is also assumed that the information retrieving system retrieves a design review material of a development plan for a device called "SYDNEY Model2000".

As shown in Fig. 16A, "SYDNEY Model2000" is input in the column of a machine type name, "DRI material" is selected within "Design evaluation" on a window 1601, and a search starting is instructed. Then, a window 1602 as shown in Fig. 16B is displayed.

Therefore, a second category "SYDNEY Model2000 Design review" is selected on this window 1602, and a search starting is instructed. As a result, a window 1603 that displays a corresponding design review is displayed as shown in Fig. 17.

Next, a procedure for retrieving an operation process association by the information retrieving system shown in Fig. 1 will be explained. Fig. 18A to Fig. 18C are diagrams that show one example of a display screen when the information retrieving system shown in Fig. 1 retrieves an operation process association. In this case, it is also assumed that the information retrieving system retrieves a design review material of a development plan for a device called "SYDNEY Model2000".

When an input frame 1810 of associative information is instructed on a window 1801 shown in Fig. 18A, ten items

of high order on the information based on the past search record including "Circuit diagram", "Parts table", and "Test specification", are displayed.

When "Test specification" is selected, a window 1802 that includes the test specification as display contents is displayed as shown in Fig. 18B. Next, when "Parts table" is selected, a window 1803 that includes a table of parts as display contents is displayed as shown in Fig. 18C.

Next, a procedure for retrieving an overview index by the information retrieving system shown in Fig. 1 will be explained. Fig. 19A and Fig. 19B are diagrams that show one example of a display screen when the information retrieving system shown in Fig. 1 retrieves an overview index. Fig. 20 is a diagram that shows one example of a hierarchical structure of documents and drawings. In this case, it is also assumed that the information retrieving system retrieves a design review material of a development plan for a device called "SYDNEY Model2000".

When "FJ1000X model2000 (F type name)" and "Test specification" are selected on a window 1901 shown in Fig. 19A, a window 1902 that includes the test specification as display contents is displayed as shown in Fig. 19B.

Specifically, this "FJ1000X model2000 (F type name)" has a hierarchical structure as shown in Fig. 20, and the test specification is one of its leaves. Therefore, when

"Test specification" has been selected on the window 1901, this hierarchical structure is traced to reach "Test specification", and the window 1902 is displayed as a result.

As described above, according to the present embodiment, in order to retrieve the design/manufacturing information registered in the plurality of web servers 110 or 120, the index server 130 is provided in the information retrieving system. Then, the register processing agent 111 delivers the abstract and the storage location of the registered information of the web servers, to the registration accept processing agent 131. Therefore, it is possible to efficiently automatically register the information registered in the web servers, into the index server 130. As a result, it is possible to enable the web servers to retrieve the information promptly and efficiently.

In the present embodiment, two web servers are used for the sake of convenience of the explanation. However, the number of the web servers is not limited to two, and it is also possible to use three or more web servers.

As explained above, according to one aspect of this invention, an abstract and storage location information of design/manufacturing information registered in the plurality of web servers, are transmitted to an index server during an idle time of the web servers. Further, the abstract

and storage location information that have been transmitted are automatically registered into the index server. Therefore, there is an effect that it is possible to obtain an information retrieving system that can efficiently automatically register the information registered in the plurality of web servers into the index server, thereby to enable information-retrieving users to promptly and efficiently retrieve the information.

Furthermore, an abstract of the design/manufacturing information is generated, and storage location information that shows a storage location of the design/manufacturing information is generated. Then, the abstract and the storage location information are transmitted to the index server during an idle time of the web servers. Therefore, there is an effect that it is possible to obtain an information retrieving system that can efficiently transmit data relating to the design/manufacturing information by utilizing the idle time of the web servers.

Furthermore, an abstract and storage location information are stored in the first repository, and the abstract and the storage location information stored in the first repository are transmitted to the index server during an idle time of the web servers. Therefore, there is an effect that it is possible to obtain an information retrieving system that can efficiently transfer the abstract

and the storage location information by using the first repository during the idle time.

Furthermore, an abstract and storage location information of the design/manufacturing information are stored in the second repository during an idle time of the index server. The consistency of the ontology of a hierarchical structure relating to the design/manufacturing information is restructured. Then, a message that the abstract and the storage location information of the design/manufacturing information have been stored in the second repository is posted to the web servers. Therefore, there is an effect that it is possible to obtain an information retrieving system that can retrieve the information based on a hierarchical structure and that can efficiently post the registration of the design/manufacturing information.

Furthermore, the information retrieving system has a bird's-eye view of the design/manufacturing information based on the abstract and the storage location information of the design/manufacturing information registered in the index server. Therefore, there is an effect that it is possible to obtain an information retrieving system that can efficiently retrieve the design/manufacturing information.

According to another aspect of this invention, an

abstract and storage location information of design/manufacturing information registered in the plurality of web servers, are transmitted to an index server during an idle time of the web servers. Further, the abstract and storage location information that have been transmitted are automatically registered into the index server. Therefore, there is an effect that it is possible to obtain an information retrieving method that can efficiently automatically register the information registered in the plurality of web servers into the index server, thereby to enable information-retrieving users to promptly and efficiently retrieve the information.

Furthermore, an abstract of the design/manufacturing information is generated, and storage location information that shows a storage location of the design/manufacturing information is generated. Then, the abstract and the storage location information are transmitted to the index server during an idle time of the web servers. Therefore, there is an effect that it is possible to obtain an information retrieving method that can efficiently transmit data relating to the design/manufacturing information by utilizing the idle time of the web servers.

Furthermore, an abstract and storage location information are stored in the first repository, and the abstract and the storage location information stored in the

first repository are transmitted to the index server during an idle time of the web servers. Therefore, there is an effect that it is possible to obtain an information retrieving method that can efficiently transfer the abstract and the storage location information by using the first repository during the idle time.

Furthermore, an abstract and storage location information of the design/manufacturing information are stored in the second repository during an idle time of the index server. The consistency of the ontology of a hierarchical structure relating to the design/manufacturing information is restructured. Then, a message that the abstract and the storage location information of the design/manufacturing information have been stored in the second repository is posted to the web servers. Therefore, there is an effect that it is possible to obtain an information retrieving method that can retrieve the information based on a hierarchical structure and that can efficiently post the registration of the design/manufacturing information.

Furthermore, the information retrieving method has a bird's-eye view of the design/manufacturing information based on the abstract and the storage location information of the design/manufacturing information registered in the index server. Therefore, there is an effect that it is

possible to obtain an information retrieving method that can efficiently retrieve the design/manufacturing information.

Although the invention has been described with respect to a specific embodiment for a complete and clear disclosure, the appended claims are not to be thus limited but are to be construed as embodying all modifications and alternative constructions that may occur to one skilled in the art which fairly fall within the basic teaching herein set forth.